On Preferred Abductive Explanations for Decision Trees and Random Forests

Gilles Audemard[1] , **Steve Bellart**[1] , **Louenas Bounia**[1] , **Frederic Koriche**[1] , **Jean-Marie Lagniez**[1] and **Pierre Marquis**[1,2]

[1]Univ. Artois, CNRS, Centre de Recherche en Informatique de Lens (CRIL), F-62300 Lens, France
[2]Institut Universitaire de France
{audemard,bellart,bounia,koriche,lagniez,marquis}@cril.fr

Abstract

Abductive explanations take a central place in eXplainable Artificial Intelligence (XAI) by clarifying with few features the way data instances are classified. However, instances may have exponentially many minimum-size abductive explanations, and this source of complexity holds even for "intelligible" classifiers, such as decision trees. When the number of such abductive explanations is huge, computing one of them, only, is often not informative enough. Especially, better explanations than the one that is derived may exist. As a way to circumvent this issue, we propose to leverage a model of the explainee, making precise her / his preferences about explanations, and to compute only *preferred explanations*. In this paper, several models are pointed out and discussed. For each model, we present and evaluate an algorithm for computing *preferred majoritary reasons*, where majoritary reasons are specific abductive explanations suited to random forests. We show that in practice the preferred majoritary reasons for an instance can be far less numerous than its majoritary reasons.

1 Introduction

Understanding predictions made by Machine Learning (ML) models is an important issue that has stimulated much research in AI for the past couple of years (see e.g., [Ribeiro *et al.*, 2016; Ribeiro *et al.*, 2018; Adadi and Berrada, 2018; Xu *et al.*, 2019; Miller, 2019; Samek *et al.*, 2019; Guidotti *et al.*, 2019; Molnar, 2020]). In this paper, we focus on the generation of *abductive explanations* for two popular families of classifiers, namely decision trees and random forests. Abductive explanations [Ignatiev *et al.*, 2019], also known as *reasons*, aim to make precise *why* a predictor classifies an input instance as positive or negative. Several types of abductive explanations exist depending on the classifier at hand. These include *prime-implicant explanations* [Shih *et al.*, 2018], also referred to as *sufficient reasons* [Darwiche and Hirth, 2020], and *majoritary reasons* [Audemard *et al.*, 2022]. Sufficient reasons and majoritary reasons coincide in the case of decision trees, but not (in general) in the case of random forests. Especially, majoritary reasons may contain redundant characteristics, while sufficient reasons are irredundant. As a counterpart, sufficient reasons are also more difficult to generate than majoritary reasons when random forests are considered [Audemard *et al.*, 2022]. Indeed, computing a sufficient reason for an instance given a random forest is intractable, while there exists a polynomial-time algorithm for generating a majoritary reason for an instance given a random forest. A computational gap can also be observed when comparing the complexity of computing a minimum-size sufficient reason to the the complexity of computing a minimum-size majoritary reason.

Abductive explanations have received much attention in the XAI literature. However, the issue of explaining why a data instance is classified as positive or negative is not entirely solved by inferring a single reason. Indeed, as shown in the following, an instance may have exponentially many reasons and even exponentially many minimum-size reasons. And this holds even if one focuses on specific reasons, such as the sufficient reasons or the majoritary reasons, and even if one considers only "intelligible" classifiers, such as decision trees. Clearly enough, when an instance has thousands of sufficient reasons, deriving their full set can be very difficult in practice. Furthermore, it does not really make sense to report a huge number of explanations to the *explainee* (the user who asked for an explanation [Miller, 2019]), since she / he will not have the cognitive capacity to consider them as a whole when too numerous.

In that case, designing alternative approaches is required. The present paper explores one of them, which is based on two research assumptions: first, not all reasons are equal, some are better than others; second, the quality of a reason does not solely rest on the reason itself, but it often depends on the explainee. Accordingly, our approach consists in exploiting *a model of the explainee, making precise her / his preferences about reasons*, to derive only *preferred reasons*. Focusing on preferred reasons has two advantages: the explanations pointed out are better (since, in essence, they are intended to match as much as possible the preferences of the explainee), and empirically their number can be drastically reduced in some cases, so that the enumeration of all preferred reasons can make sense in situations when enumerating all possible reasons would be out of reach.

In our study, several models are proposed. We start with

a simple model leading to dichotomous preferences over reasons, where "good" ones are expected to be based only on a specific subset of features. This is enough to prevent from deriving explanations based on features which are not intelligible or actionable, or those containing protected characteristics, possibly reflecting a biased decision. We also present a model where "good" reasons are those satisfying a predefined constraint. The next step is to consider more elaborated preferences, which are not dichotomous in essence but are more gradual. This is achieved first by considering an ordinal preference relation, having the form of a prioritization of the features. Such a preference relation is suited to scenarios where some features are considered as more expected than others in explanations, while any compensation between features from distinct strata is forbidden. Finally, a last model is presented, based on a linear (dis)utility / cost function over the features; here, every feature has a weight, and weights are aggregated in an additive way. This leads to a cardinal preference relation over reasons.

For all those models, deriving a preferred sufficient reason given a random forest is intractable (simply because this is already the case when no preference model is considered). Contrastingly, for each of the four preference models considered in the paper, but the last one, we present a polynomial-time algorithm that infers a preferred majoritary reason for an input data instance given a random forest. Though the problem of deriving a preferred majoritary reason for an instance given a random forest and a weight mapping is NP-hard in general, we show how one can leverage a WEIGHTED PARTIAL MAXSAT solver to compute it. Finally, we present the results of an empirical evaluation illustrating the benefits that can be achieved in practice by leveraging a model of the user preferences in the computation of majoritary reasons.

The rest of the paper is organized as follows. We start with some preliminaries (Section 2) where the notions of decision tree and of random forest are recalled. Abductive explanations, including sufficient reasons and majoritary reasons for random forests, are presented in Section 3. Preference models and algorithms for deriving preferred majoritary reasons are provided in Section 4. Empirical results are reported in Section 5, before the concluding section (Section 6). A full-proof version of the paper is available at www.cril.univ-artois.fr/expekctation/papers.html.

2 Decision Trees and Random Forests

For an integer n, let $[n] = \{1, \cdots, n\}$. By $\mathcal{F}_n$ we denote the class of all Boolean functions from $\{0,1\}^n$ to $\{0,1\}$, and we use $X_n = \{x_1, \cdots, x_n\}$ to denote the set of input Boolean variables. Any Boolean vector $\boldsymbol{x} \in \{0,1\}^n$ is called an *instance*. For any function $f \in \mathcal{F}_n$, an instance $\boldsymbol{x} \in \{0,1\}^n$ is called a *positive example* of f if $f(\boldsymbol{x}) = 1$, and a *negative example* if $f(\boldsymbol{x}) = 0$.

We refer to f as a propositional formula when it is described using the Boolean connectives $\wedge$ (conjunction), $\vee$ (disjunction) and $\neg$ (negation), together with the variables from X_n, and the constants 1 (true) and 0 (false). The set of variables occurring in a formula f is denoted $Var(f)$. As usual, a *literal* ℓ_i is a variable x_i or its negation $\neg x_i$, also denoted $\overline{x}_i$. For the literals x_i and $\neg x_i$, we note $var(x_i) = var(\neg x_i) = x_i$. A *term* t is a conjunction of literals, and a *clause* c is a disjunction of literals. A DNF *formula* is a disjunction of terms and a CNF *formula* is a conjunction of clauses. In the following, we shall often treat instances as terms, and terms or clauses as sets of literals. For an assignment $\boldsymbol{z} \in \{0,1\}^n$, the corresponding term is

$$t_{\boldsymbol{z}} = \bigwedge_{i=1}^{n} x_i^{z_i} \text{ where } x_i^0 = \overline{x}_i \text{ and } x_i^1 = x_i$$

For a subset of variables S and a term t, we use $t[S]$ to denote $\{\ell \in t : var(\ell) \in S\}$. For an assignment $\boldsymbol{x}$ and a clause c, we denote by $c[\boldsymbol{x}]$ the clause $c \cap t_{\boldsymbol{x}}$. A term t *covers* an assignment $\boldsymbol{x}$ if $t \subseteq t_{\boldsymbol{x}}$. An *implicant* of a Boolean function f is a term t such that $f(\boldsymbol{x}) = 1$ for every assignment $\boldsymbol{x}$ covered by t. A *prime implicant* of f is an implicant t of f such that no proper subset of t is an implicant of f.

With these basic notions in hand, a (Boolean) *decision tree* [Breiman *et al.*, 1984; Quinlan, 1986] on X_n is a binary tree T, each of whose internal nodes is labeled with one of n input variables, and whose leaves are labeled 0 or 1. Without loss of generality, every variable is supposed to occur at most once on any root-to-leaf path. The value $T(\boldsymbol{x})$ of T on an input instance $\boldsymbol{x}$ is given by the label of the leaf reached from the root as follows: at each node go to the left or right child depending on whether the input value of the corresponding variable is 0 or 1, respectively. It is well-known that a decision tree T can be encoded in linear time into an equivalent CNF formula $\texttt{CNF}(T)$, where the clauses in $\texttt{CNF}(T)$ are precisely the negations of the terms describing the 0-paths of T (i.e., the root-to-leaf paths of T ending with 0-leaves).

A (Boolean) *random forest* [Breiman, 2001] on X_n is an ensemble $F = \{T_1, \cdots, T_m\}$, where each T_i ($i \in [m]$) is a decision tree on X_n, and such that the value $F(\boldsymbol{x})$ is given by

$$F(\boldsymbol{x}) = \begin{cases} 1 & \text{if } \frac{1}{m}\sum_{i=1}^{m} T_i(\boldsymbol{x}) > \frac{1}{2} \\ 0 & \text{otherwise.} \end{cases}$$

The size of F is given by $|F| = \sum_{i=1}^{m} |T_i|$, where $|T_i|$ is the number of nodes occurring in T_i. The class of decision trees on X_n is denoted $\texttt{DT}_n$, and the class of random forests with at most m decision trees (with $m \geq 1$) over $\texttt{DT}_n$ is denoted $\texttt{RF}_{n,m}$. Finally, $\texttt{RF}_n$ is the union of all $\texttt{RF}_{n,m}$ for $m \geq 1$. Obviously enough, whenever $F = \{T\}$, we have $F(\boldsymbol{x}) = T(\boldsymbol{x})$ for every $\boldsymbol{x} \in \{0,1\}^n$.

3 On Abductive Explanations

An important issue in XAI is to develop approaches to explain *why* a predictor classifies some data instance as positive or negative. This calls for a notion of *abductive explanation* for an instance [Ignatiev *et al.*, 2019].[1] Specifically, an *abductive explanation* (also known as a *reason*) for an instance $\boldsymbol{x} \in \{0,1\}^n$ given a Boolean function $f \in \mathcal{F}_n$ is a term t that

[1] Unlike [Ignatiev *et al.*, 2020], we do not require abductive explanations to be minimal w.r.t. set inclusion. Our abductive explanations correspond to so-called weak abductive explanations in [Huang *et al.*, 2021].

covers $\boldsymbol{x}$ and is an implicant of f (resp. $\neg f$) when $f(\boldsymbol{x}) = 1$ (resp. $f(\boldsymbol{x}) = 0$).

Since the predictor itself f (and not a proxy for it) is considered in this definition, the abductive explanations t for $\boldsymbol{x}$ given f are *guaranteed to be correct* in the sense that every instance $\boldsymbol{x}'$ covered by t is provably classified by f in the same way as $\boldsymbol{x}$. This property enables the explainee to reason from the abductive explanations that are reported. Accordingly, abductive explanations heavily differ from explanations computed using model-agnostic approaches (see [Ignatiev, 2020]), for which the previous correctness condition does not hold in general.

Clearly enough, an abductive explanation for $\boldsymbol{x}$ given f always exists, since $t = t_{\boldsymbol{x}}$ is such a (trivial) explanation. However, it may contain many features ℓ that are useless, in the sense that the instance that coincides with $\boldsymbol{x}$ except on ℓ is classified in the same way as $\boldsymbol{x}$. In such a case, $t_{\boldsymbol{x}} \setminus \{\ell\}$ must be preferred to $t_{\boldsymbol{x}}$ for explaining the way $\boldsymbol{x}$ has been classified by f. Pushing this idea a step further, one gets the notion of *sufficient reason* for $\boldsymbol{x}$ given f [Darwiche and Hirth, 2020].

Definition 1 *Let $f \in \mathcal{F}_n$ be a Boolean function and $\boldsymbol{x} \in \{0,1\}^n$ be an instance. A* sufficient reason *for $\boldsymbol{x}$ given f is a term t that covers $\boldsymbol{x}$ and is a prime implicant[2] of f (resp. $\neg f$) if $f(\boldsymbol{x}) = 1$ (resp. $f(\boldsymbol{x}) = 0$).*

The complexity of identifying sufficient reasons and of deriving one of them differs with the representation of f. Thus, the problem of deciding whether a given term is a sufficient reason t for an input instance $\boldsymbol{x} \in \{0,1\}^n$ given a decision tree T can be solved in polynomial time since it amounts to testing whether t is an implicant of T, and for all $\ell \in t$, $t \setminus \{\ell\}$ is no longer an implicant of T. Contrastingly, this problem becomes intractable when the classifier under consideration is a random forest $F \in \mathtt{RF}_n$ (it has been shown DP-complete in this case [Izza and Marques-Silva, 2021]).

Introduced in [Audemard *et al.*, 2022], *majoritary reasons* are abductive explanations that are specific to random forests.

Definition 2 *Let $F = \{T_1, \ldots, T_m\}$ be a random forest in $\mathtt{RF}_{n,m}$ and $\boldsymbol{x} \in \{0,1\}^n$ be an instance. A* majoritary reason *for $\boldsymbol{x}$ given F is a term t covering $\boldsymbol{x}$, such that t is an implicant of at least $\lfloor \frac{m}{2} \rfloor + 1$ decision trees T_i (resp. $\neg T_i$) if $F(\boldsymbol{x}) = 1$ (resp. $F(\boldsymbol{x}) = 0$), and for every $\ell \in t$, $t \setminus \{\ell\}$ does not satisfy this last condition.*

Majoritary reasons are abductive explanations since if a term t implies a majority of decision trees in F, then it is an implicant of F. However, the converse implication does not hold and majoritary reasons and sufficient reasons do not coincide in general (majoritary reasons may contain redundant literals). However, in the restricted case when $F = \{T\}$, the majoritary reasons of any $\boldsymbol{x}$ given F coincide with its sufficient reasons.

What makes majoritary reasons valuable is that they can be generated in linear time using the following *greedy algorithm*. For the case when $F(\boldsymbol{x}) = 1$, start with $t = t_{\boldsymbol{x}}$, and iterate over the literals ℓ of t by checking whether t deprived of ℓ is an implicant of at least $\lfloor \frac{m}{2} \rfloor + 1$ decision trees of F. If so, remove ℓ from t and proceed to the next literal. Once all literals in $t_{\boldsymbol{x}}$ have been examined, the final term t is by construction an implicant of a majority of decision trees in F, such that removing any literal from it would lead to a term that is no longer an implicant of this majority. So, t is by construction a majoritary reason. The case where $F(\boldsymbol{x}) = 0$ is similar, by simply replacing in F each T_i with its negation (which can be easily obtained by replacing the 0-leaves of T_i by 1-leaves, and vice-versa). Indeed, the resulting forest is equivalent to $\neg F$, thus it classifies $\boldsymbol{x}$ as a positive instance precisely when $F(\boldsymbol{x}) = 0$ (see Proposition 1 from [Audemard *et al.*, 2022] for details). This simple greedy algorithm runs in $\mathcal{O}(n|F|)$ time, using the fact that, on each iteration, checking whether t is an implicant of T_i (for each $i \in [m]$) can be done in $\mathcal{O}(n|T_i|)$ time. Thus, the generation of a single majoritary reason for $\boldsymbol{x}$ given F is tractable.

Unfortunately, this tractability result cannot be extended to the case when all majoritary reasons are looked for, simply because majoritary reasons can be too numerous. Thus, an instance may have exponentially many majoritary reasons given a random forest. Notably, this issue does not come from the specific nature of majoritary reasons, that may contain redundant literals. Indeed, an instance may also have exponentially many sufficient reasons given a random forest. To be more precise, even in the restricted case when the forest reduces to a single tree, an instance may have exponentially many minimum-size majoritary reasons (or, equivalently, exponentially many minimum-size sufficient reasons since the two notions coincide for decision trees).

Proposition 1 *For every $n \in \mathbb{N}$ such that n is odd, there is a decision tree $T \in \mathtt{DT}_n$ of depth $\frac{n+1}{2}$ such that T contains $2n+1$ nodes and there is an instance $\boldsymbol{x} \in \{0,1\}^n$ such that the number of minimum-size sufficient reasons for $\boldsymbol{x}$ given T is equal to $2^{\sqrt{n-1}}$.*

Experiments have shown that such worst-case results are not rare in practice. When thousands of reasons or much more exist, computing each of them can be out of reach, and even when this computation is feasible, the user who asks for explanations (aka the explainee) cannot handle them as a whole due to their cognitive limitations. In such a situation, computing a single reason (or only a few reasons) is not a panacea since reasons may differ a lot one another (reasons can be pairwise disjoint). Thus, it can easily be the case that the reason that is computed is not good enough for the explainee, and that a much better reason actually exists but is not the one that has been reported.

4 On Preferred Abductive Explanations

A rational way to deal with this issue consists in focusing on a subset of reasons, the so-called *preferred ones*. Defining what is a "preferred" or "sufficiently good" explanation is a difficult task in general. There is no consensus about it [Doshi-Velez and Kim, 2017; Lipton, 2018; Narayanan *et al.*, 2018; Srinivasan and Chander, 2020], because in the general case several criteria must be taken into account to assess the quality of an explanation. Furthermore, some of these criteria are intrinsic to explanations (e.g., considering minimum-size explanations), but others heavily depend on the explainee. To

[2] This explains why sufficient reasons are also known as prime-implicant explanations [Shih *et al.*, 2018].

focus only on "preferred" or at least on "sufficiently good" explanations, a formal model of the explainee must be designed and leveraged. Such a model of the explainee can be more or less sophisticated and take various forms.

We present several models in the following, and we define notions of preferred reasons based on them. Those preferred reasons can be considered w.r.t. the full set of abductive explanations, or to subsets of it, especially those containing only sufficient reasons, or those containing only majoritary reasons. Though the notions of preferred reasons make sense for any Boolean classifier, our results are mainly about random forests since they concern majoritary reasons.

4.1 Dichotomous Preferences over Explanations

We start with two models where the preferences of the explainee are *dichotomous*, i.e., reasons can be partitioned into two sets: the one containing reasons that are "sufficiently good" and the other one containing reasons judged as "not good enough".

Focusing on specific features. A very simple explainee model consists of a subset $S \subseteq X_n$ of features. This model is enough to handle a couple of situations of interest where the explainee wants to discard explanations that refer to *non-understandable concepts* (modeled as features outside S). Such explanations to be discarded may contain features corresponding to quite technical notions, that are not understood by the explainee (e.g., a medical term for a patient who is not a physician), possibly because they are not documented or are quite vague in essence (consider for instance, feature `Other` (`O`) in the *compas* dataset).

Definition 3 *Let $f \in \mathcal{F}_n$, $S \subseteq X_n$, and $\boldsymbol{x} \in \{0,1\}^n$. A* reason built upon S *for $\boldsymbol{x}$ given f is a reason t for $\boldsymbol{x}$ given f such that $Var(t) \subseteq S$.*

Ensuring that only explanations built upon features in S are generated is also helpful for ensuring other objectives. Thus, the presence of some *protected features* should be avoided in explanations whenever this is possible, since the impossibility to let such features aside precisely reflects the fact that the decision made was biased [Darwiche and Hirth, 2020]. For instance, in a college admission problem, consider an applicant for which the decision made by the classifier is positive: if every abductive explanation of this decision mentions the fact that the applicant comes from a rich hometown (a protected feature), the decision is biased. Hence "coming from a rich hometown" should not belong to S. Beyond understandability or bias issues, the absence of features that are *not actionable* must be avoided in explanations. Being not actionable simply means that one cannot (or one can hardly) change their values. For instance, in a loan classification problem, if an abductive explanation that a loan has not been granted to an applicant mentions that he/she is over fifty years old, another explanation should be looked for. Indeed, the applicant cannot change it. In this scenario, the fact that the applicant is over fifty years old should not belong to S.

Interestingly, deciding whether explanations built solely upon such a pre-specified set S of features exist can be easily achieved when considering sufficient reasons given decision trees, and more generally majoritary reasons given random forests.

Proposition 2 *Let $F \in \mathtt{RF}_n$ and $\boldsymbol{x} \in \{0,1\}^n$. For any set $S \subseteq X_n$, deciding whether a majoritary reason built upon S for $\boldsymbol{x}$ given F exists, and deriving such a reason when this is the case, can be done in $\mathcal{O}(n|F|)$ time using a greedy algorithm.*

Requiring constraints to be satisfied. A slightly more complex model of the explainee takes the form of a formula C (a constraint over X_n) that every explanation must satisfy in order to be acceptable by the explainee. For instance, such a constraint C may reflect some regulation statement that must be obeyed.

Definition 4 *Let $f \in \mathcal{F}_n$, C a formula over X_n, and $\boldsymbol{x} \in \{0,1\}^n$. A* reason satisfying C *for $\boldsymbol{x}$ given f is a reason t for $\boldsymbol{x}$ given f such that $t \models C$.*

This time again, under some assumptions on C, deciding whether explanations satisfying C exist is computationally easy when considering sufficient reasons given decision trees, and more generally majoritary reasons given random forests.

Proposition 3 *Let $F \in \mathtt{RF}_n$ and $\boldsymbol{x} \in \{0,1\}^n$. Let C be a formula over X_n. Deciding whether a majoritary reason for $\boldsymbol{x}$ given F that implies C exists can be done in $\mathcal{O}(n + |F|)$ time, and deriving such a reason when this is the case, can be done in $\mathcal{O}(n|F|)$ time using a greedy algorithm when C belongs to propositional fragment offering a polynomial-time implicant test (e.g., C is a* `CNF` *formula).*

4.2 More Gradual Preferences over Explanations

The two previous models induce dichotomous preferences over explanations. While such a separation in two classes is convenient in some cases, one would expect *more graduality* in other cases, in order (for instance) to avoid the presence of some features in explanations without forbidding it.

Inclusion-preferred explanations. Here is an approach to define preference relations that are typically not dichotomous. Consider a total preorder $\leq$ over X_n, such that $x_i \leq x_j$ means that feature x_i is considered as at most as important or as at most as expected as feature x_j. $\leq$ can be represented by a *prioritization* (or stratification) of X_n, i.e., an ordered partition $S_1, \ldots, S_p$ of the features from X_n such that x_i and x_j belongs to the same set of the partition if and only if $x_i \leq x_j$ and $x_j \leq x_i$, and S_i precedes S_j in the partition (i.e., $i < j$) whenever every element of S_i is before every element of S_j w.r.t. $<$.

Based on the prioritization $S_1, \ldots, S_p$ of X_n induced by $\leq$, one can define a preference relation $\sqsubset$ on the terms over X_n by stating that $t \sqsubset t'$ if and only if $\exists i \in \{1, \ldots, p\} \forall j \in \{1, \ldots, i-1\}$, $t[S_j] = t'[S_j]$ and $t[S_i] \subset t'[S_i]$.[3] It can be checked that $\sqsubset$ is a strict, partial order (i.e., an irreflexive and transitive relation). Given a set of terms, the minimal ones w.r.t. $\sqsubset$ correspond intuitively to those that contain as few unexpected literals as possible (where the comparison is based on set inclusion). On this ground, we are now ready to define the notion of inclusion-preferred reason:

[3] The construction is reminiscent to the one used for characterizing preferred subtheories in [Brewka, 1989].

Definition 5 *Let $f \in \mathcal{F}_n$, $\leq$ a total preorder over X_n, and $\boldsymbol{x} \in \{0,1\}^n$. An* inclusion-preferred reason *for $\boldsymbol{x}$ given f is a reason t for $\boldsymbol{x}$ given f such that there is no reason t' for $\boldsymbol{x}$ given f satisfying $t' \sqsubset t$.*

The greedy algorithm presented above for generating majoritary reasons can be cast in such a way that it generates inclusion-preferred majoritary reasons:

Proposition 4 *Let $F \in \mathtt{RF}_n$, $\leq$ be a total preorder over X_n, and $\boldsymbol{x} \in \{0,1\}^n$. Deriving an inclusion-preferred majoritary reason for $\boldsymbol{x}$ given F and $\leq$ can be done in $\mathcal{O}(|G| + n|F|)$ time where G is the graph $(X_n, \leq)$.*

Prioritizations as those considered here are available in a number of contexts. For instance, features can be ordered by aggregating the frequencies of the words used in their descriptions. Assuming that for most of the explainees rare words are less understood than frequent words, it makes sense to order the features by comparing the less frequent words used in their descriptions. Some resources can be exploited to this end, for instance the *wordfreq* library (pypi.org/project/wordfreq/). In wordfreq's wordlists, meaningless precision is avoided by packing the words into frequency bins (i.e., building a prioritization of the words). Thus, all words having the same Zipf frequency[4] rounded to the nearest hundredth are considered to have the same frequency.

Minimum-weight explanations. Another very standard way to model a preference relation over a combinatorial domain is to take advantage of a *(dis)utility function* (or a cost function). In our context, this amounts to associating a (dis)utility value (a weight) with every feature. Such a weight indicates how much the corresponding feature is not expected to occur in an explanation (so the lower the better). Then the (dis)utility (cost) of an explanation is calculated as the sum of the weights of the features in it. This time, the resulting preference relation is a total preorder over the explanations, the best explanations being those of minimal cost. Note that the presence of many "cheap" features in an explanation can be balanced by the presence of a a single "expensive" feature.

Definition 6 *Let $f \in \mathcal{F}_n$. Let $w : X_n \to \mathbb{N}^*$ be a weight mapping associating with every feature a positive integer. A* minimum-weight reason *for $\boldsymbol{x}$ given f and w is a reason t for $\boldsymbol{x}$ given f that minimizes $\Sigma_{x \in Var(t)} w(x)$.*

In order to compute a minimum-weight majoritary reason, unlike what was done before, one cannot take advantage of any polynomial-time variant of the greedy algorithm for deriving majoritary reasons. Indeed, in the general case, the computation of a minimum-weight majoritary reason is NP-hard in the broad sense. This comes from the facts that (1) a minimum-size majoritary reason t for an instance given a random forest is a minimum-weight majoritary reason t for an instance given a random forest and a weight mapping w_1 such that for every $i \in [n]$, $w_1(x_i) = 1$, and (2) that deriving a minimum-size majoritary reason t for an instance given a random forest is NP-hard [Audemard *et al.*, 2022].

[4]The Zipf frequency of a word is the base-10 logarithm of the number of times it appears per billion words.

Nevertheless, one can generalize the approach presented in [Audemard *et al.*, 2022] for computing minimum-size majoritary reasons to the case of minimum-weight majoritary reasons. Basically, this amounts to solving an instance of the WEIGHTED PARTIAL MAXSAT problem. Such an instance consists of a pair $(C_{\text{soft}}, C_{\text{hard}})$ where C_{soft} and C_{hard} are (finite) sets of weighted clauses. A weighted clause is an ordered pair (c, w) where w is a natural number or ∞. w gives the cost of falsifying c. If w is infinite, the clause is hard, otherwise it is soft. The goal is to find a Boolean assignment that maximizes the sum of the weights of the clauses c in C_{soft} that are satisfied, while satisfying all clauses c such that $(c, \infty) \in C_{\text{hard}}$.

Proposition 5 *Let $F = \{T_1, \ldots, T_m\}$ be a random forest in $\mathtt{RF}_{n,m}$ and $\boldsymbol{x} \in \{0,1\}^n$ be an instance such that $F(\boldsymbol{x}) = 1$. Let $w : X_n \to \mathbb{N}^*$ be a weight mapping. A minimum-weight majoritary reason for $\boldsymbol{x}$ given F and w is given by $t_{\boldsymbol{x}} \cap t_{\boldsymbol{v}^*}$, where $\boldsymbol{v}^*$ is a solution of the instance $(C_{\text{soft}}, C_{\text{hard}})$ of the* WEIGHTED PARTIAL MAXSAT *problem such that:*

$$C_{\text{soft}} = \{(\overline{x_i}, w(x_i)) : x_i \in t_{\boldsymbol{x}}\} \cup \{(x_i, w(x_i)) : \overline{x}_i \in t_{\boldsymbol{x}}\}$$
$$C_{\text{hard}} = \{(\overline{y_j} \vee c[\boldsymbol{x}], \infty) : i \in [m], c \in \mathtt{CNF}(T_i)\} \cup \mathtt{CNF}\left(\sum_{i=1}^{m} y_i > \frac{m}{2}\right)$$

where $\{y_1, \ldots, y_m\}$ are fresh variables, and $\mathtt{CNF}(\sum_{i=1}^{m} y_i > \frac{m}{2})$ is a $\mathtt{CNF}$ encoding of the constraint $\sum_{i=1}^{m} y_i > \frac{m}{2}$.

In the case when $F(\boldsymbol{x}) = 0$, it is enough to consider the same instance of WEIGHTED PARTIAL MAXSAT as above, except that each T_i ($i \in [m]$) is replaced by $\neg T_i$.

Thanks to Proposition 5, one can leverage solvers that have been designed so far for solving instances of WEIGHTED PARTIAL MAXSAT (see e.g. [Martins *et al.*, 2014; Ansótegui and Gabàs, 2017]) in order to compute minimum-weight majoritary reasons. Interestingly, one can also easily ensure that the minimum-weight majoritary reasons that are computed are (A) built upon a specific set S of features and / or (B) satisfy a given $\mathtt{CNF}$ constraint C. To do so, it is enough to add some weighted clauses to C_{hard} before computing $\boldsymbol{v}^*$, namely (A) $(\overline{\ell}, \infty)$ for every $\ell \in t_{\boldsymbol{x}}$ such that $var(\ell) \notin S$ and (B) (c, ∞) for every $c \in C$. Note however that there is no guarantee that the minimum-weight majoritary reason that is computed is a minimum-size minimum-weight majoritary reason. In addition, because of (B), we can easily leverage the WEIGHTED PARTIAL MAXSAT solver used to compute a single minimum-weight majoritary reason to enumerate the minimum-weight majoritary reasons (each time such a reason t has been computed, it is enough to add the hard clause $(\neg t, \infty)$ to C_{hard} in order to block the further generation of t).

Finally, whenever computing a single minimum-weight majoritary reason proves hard, we can take advantage of an *anytime* WEIGHTED PARTIAL MAXSAT solver (like `openwbo` [Martins *et al.*, 2014]) to derive an *approximation* of such a reason. Using an anytime solver, solutions $\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots$ are successively generated. All those solutions satisfy C_{hard}, their respective weights do not increase, and the sequence

converges towards an optimal solution $\boldsymbol{v}^*$. Thus, the last element $\boldsymbol{v}_k$ of the sequence that has been derived at the time when the solver is stopped can be used to generate a majoritary reason that can be viewed as an approximation of a minimum-weight majoritary reason. To do so, $t_{\boldsymbol{v}_k}$ just needs to be post-processed using the greedy algorithm so as to remove redundant literals from it, thus ensuring that the resulting term is a majoritary reason for $\boldsymbol{x}$ given F. Note that there is no guarantee that the weight of this term is close to the weight of $\boldsymbol{v}^*$.

5 Experiments

Experimental setup. We have considered 22 datasets for binary classification, which are standard benchmarks from the repositories Kaggle (www.kaggle.com), OpenML (www.openml.org), and UCI (archive.ics.uci.edu/ml/), and we have learned random forests from them. Some of these datasets are listed in Table 1. "name" is the name of the dataset. "short description" indicates in a few words the goal of the prediction. "#instance" is the number of instances in the dataset, "#feature" is the number of features used to describe the instances, and "#class" is the number of classes. Finally, "source" makes precise the repository the dataset comes from.

In the computation of random forests, categorical features have been treated as arbitrary numbers. Numeric features have been binarized on-the-fly by the random forest learning algorithm we used, namely version 0.23.2 of the Scikit-Learn library [Pedregosa *et al.*, 2011]. All hyper-parameters of the learning algorithm have been set to their default value, except the number of trees. This parameter has been tuned to ensure that the accuracy of the forest is good enough. For each dataset b, a 10-fold cross validation process has been achieved. For each dataset b and each random forest F for b among the 10 that have been learned, 25 instances have been picked up in the dataset, leading to a pool of 250 instances per dataset (unless the dataset contains less than 250 instances, in which case the pool consists of the entire dataset).

In our experiments, we have focused on the computation of minimum-weight majoritary reasons. Indeed, minimum-weight majoritary reasons is the only type of preferred reasons (among those considered in the paper) which is not guaranteed to be practical (for each of the other types of preferred majoritary reasons described here, a polynomial-time greedy algorithm exists). For each dataset b, we generated all the minimum-weight majoritary reasons for the instances $\boldsymbol{x}$ in the pool associated with b, up to exhaustion or a time limit of 60s per instance. Each time a minimum-weight majoritary reason for the instance $\boldsymbol{x}$ at hand has been generated, this reason has been blocked for not being computed twice, and the computation resumed.

Since no weight functions are primarily associated with the datasets used in our experiments (remember that user preferences are not intrinsic to the datasets), we have considered as a second best for each dataset (1) the uniform weight function, where each feature has weight 1 (in that case, the minimum-weight majoritary reasons are precisely the minimum-size ones), (2) for each random forest F, the opposite of the SHAP score [Lundberg and Lee, 2017; Lundberg *et al.*, 2020] of each feature of the instance $\boldsymbol{x}$ at hand given F computed using SHAP (shap.readthedocs.io/en/latest/api.html), (3) the opposite of the f-importance of each feature in F as computed by Scikit-Learn [Pedregosa *et al.*, 2011], and (4) the opposite of the Zipf frequency of each feature viewed as a word in the *wordfreq* library. In case (2) the weights that are computed are local ones, i.e., they depend on the instance $\boldsymbol{x}$ that is considered. This contrasts with the weights computed in cases (3) (measuring the global importance of features) and (4). While (2) and (3) aim to favor majoritary reasons involving the most important features from the explainability side, (4) emphasizes intelligibility.

Most of the time, the available feature weights are not guaranteed to be positive integers (as for the weights above, except in case (1)). When negative weights must be taken into account, the weight mapping w is first updated into $w - min_{x_i \in X_n} w(x_i) + 1$; doing so, the weight of every feature is made positive but the induced ordering over explanations is preserved so that the minimum-weight majoritary reasons do not change. When fractional (yet positive) weights are considered, one changes w into $10^k \cdot w$ where k is the maximum number of digits after the decimal point in the representations of the current weights; that way, all weights are turned into integers; again, the the induced ordering over explanations is preserved so that the minimum-weight majoritary reasons are kept. Indeed, it is well-known that a (dis)utility (or cost) function that can be subjected to a positive affine transformation without altering the implied preference order.

Since the weight functions are intended to be part of the input, we did not count the computation time required to generate them within the 60s. We took advantage of the anytime WEIGHTED PARTIAL MAXSAT `openwbo` [Martins *et al.*, 2014] to derive minimum-weight majoritary reasons. All the experiments have been conducted on a computer equipped with Intel(R) Core(TM) i9-9900 CPU @ 3.10GHz - 16 cores and 64 GB of memory.

Experimental results. Table 2 reports an excerpt of the results, based on 14 datasets. For each dataset and the random forests learned from it, the table gives the name of the dataset (name), the mean accuracy (%A) of the forests, the mean number of binary features (#B) in them, and the number of instances in the pool (#I). Then for each dataset b and each weight function type used, it gives the number of instances $\boldsymbol{x}$ in the pool for which at least one (1) or all (A) minimum-weight majoritary reasons for $\boldsymbol{x}$ have been derived in less than 60s; column (nb) gives the mean number (and the standard deviation) of minimum-weight majoritary reasons that have been obtained for the instances for which every minimum-weight majoritary reason has been computed.

The empirical results clearly show that computing preferred majoritary reasons is feasible in practice. Indeed, for many datasets and instances, all minimum-weight majoritary reasons have been computed within 60s whatever the weight function type used. We can also observe that the number of instances for which all minimum-weight majoritary reasons have been computed is often close to the number of instances for which at least one minimum-weight majoritary reason has been derived within the allocated time period. Furthermore,

name	short description	(#instance #feature #class)	source
divorce	predict whether couples will divorce or not	(170, 54, 2)	Kaggle
compas	determine whether defendants will re-offend or not over a two-year period	(5278, 14, 2)	OpenML
employee	determine whether employees will leave or not in the 2-next years	(4653, 8, 2)	UCI
student mat	predict whether students will succeed or failin mathematics	(395, 32, 2)	UCI
student por	predict whether students will succeed or fail in Portuguese	(649, 32, 2)	UCI
anneal 2	predict about annealing, a heat treatment used in metallurgy	(898, 38, 2)	UCI
placement	predict about student placement	(215, 13, 2)	Kaggle
heart	predict the presence or the absence of heart disease	(303, 13, 2)	OpenML
diabetes	predict whether patients are diabetic or not	(768, 8, 2)	Kaggle
horse	predict whether horses can survive or not	(299, 27, 2)	UCI
indian liver patient	classify patients with liver disease or no disease	(583, 10, 2)	UCI
banknote	determine whether banknotes are genuine or not	(1372, 4, 2)	Kaggle
startup	predict whether startups will succeed or fail	(923, 45, 2)	Kaggle
farm-ads	decide whether owners will approve advertisements or not	(1543, 54877, 2)	UCI

Table 1: Some of the datasets used in our experiments.

dataset / random forest				minimum-size			SHAP			f-importance			wordfreq		
name	%A	#B	#I	l	A	nb	l	A	nb	l	A	nb	l	A	nb
divorce	97.65	50	170	170	161	41.6 (±77.4)	169	169	1.2 (±0.4)	170	170	1.1 (±0.3)	170	170	1.0 (±0.1)
compas	66.51	65	250	250	250	6.0 (±9.9)	249	249	1.9 (±1.8)	247	243	2.7 (±3.9)	250	250	2.4 (±2.7)
employee	83.17	72	250	243	174	8.9 (±12.6)	243	235	2.0 (±2.1)	249	245	2.1 (±3.5)	239	204	4.4 (±7.6)
student mat	90.63	144	250	250	217	44.7 (±60.4)	250	250	1.1 (±0.2)	250	250	1.1 (±0.3)	250	250	1.2 (±0.4)
student por	91.99	171	250	19	10	43.0 (±38.4)	16	16	1.1 (±0.3)	14	14	1.4 (±0.8)	25	24	1.4 (±0.7)
anneal 2	99.11	203	250	250	200	26.8 (±39.0)	240	240	1.1 (±0.3)	241	240	1.1 (±0.3)	248	248	1.1 (±0.4)
placement	93.55	262	215	215	145	50.7 (±61.0)	215	215	1.4 (±1.3)	215	213	1.3 (±0.7)	215	212	1.2 (±0.6)
heart	78.3	263	250	250	236	41.7 (±59.5)	250	250	1.3 (±0.6)	250	250	1.3 (±0.6)	250	250	1.4 (±0.9)
diabetes	72.28	433	250	250	248	18.2 (±37.7)	250	250	1.3 (±1.1)	250	250	1.3 (±0.7)	250	250	1.1 (±0.4)
horse	87.31	540	250	62	6	72.7 (±46.0)	50	50	1.4 (±0.8)	55	55	1.4 (±0.8)	42	41	1.4 (±0.6)
ind. l. pat.	69.61	613	250	250	187	54.1 (±54.3)	250	250	1.6 (±1.7)	250	250	1.5 (±0.9)	250	250	2.3 (±2.8)
banknote	99.42	652	250	190	37	11.1 (±7.1)	215	207	2.0 (±2.7)	237	231	1.8 (±1.8)	160	150	1.2 (±0.5)
startup	80.18	3517	250	57	0	- (-)	40	38	1.4 (±0.7)	43	42	1.5 (±0.9)	43	38	1.8 (±1.1)
farm-ads	87.3	5389	250	25	0	- (-)	250	250	1.0 (±0.0)	11	11	1.0 (±0.0)	25	25	1.2 (±0.4)

Table 2: Some statistics about the computation of minimum-weight majoritary reasons for instances from some datasets.

it turns out that the use of weight function types (2), (3), and (4) had a drastic effect on the number of reasons, and has thus favored the computation of all minimum-weight majoritary reasons by reducing their number. Finally, it is worth noting that for each dataset b (including the "harder ones", e.g., *farm-ads*), each instance in the pool of b, and each weight function type (1) to (4), one has been able to compute in less than 60s an approximation of a minimum-weight majoritary reason. More detailed statistics about the computation time, the number of reasons that have been generated and their size have been drawn for each dataset; they are available at www.cril.univ-artois.fr/expekctation/.

6 Conclusion

In this paper, we have considered the problem of generating abductive explanations (alias reasons), where an abductive explanation for an input instance given a predictor aims to make precise why the predictor classifies the instance as positive or negative. It turns out that an instance may have exponentially many reasons and even exponentially many minimum-size reasons, even for "intelligible" ML models such as decision trees. An exponential number of minimum-size reasons can also be obtained for specific classes of reasons, especially the sufficient reasons or the majoritary reasons, where majoritary reasons are abductive explanations suited to random forests that, unlike sufficient reasons, are not guaranteed to be irredundant but can be generated in a tractable way. Because the enumeration of all the reasons is out of reach (and not desirable), it is important to be able to focus on some reasons, those that are considered as "good enough" by the explainee. This calls for defining preference models and leveraging them to derive only preferred reasons.

To this purpose, we have presented four preference models, analyzed the complexity of computing a preferred majoritary reason for each of them, and explained how to do so. Beyond leading to reasons that better fit the user expectations, experiments have shown that the exploitation of user preferences may drastically reduce the number of reasons, rendering their enumeration possible in situations where computing all majoritary reasons would be infeasible.

Taking advantage of user preferences, as done in the paper, is a first step in the direction of characterizing the explanations that the explainee expects. To go a step further and achieve an evaluation of explanations that is not just functionally-grounded, but also human-grounded or even application-grounded (to borrow the words used in [Doshi-Velez and Kim, 2017]), one would need a human expert to be available (or, alternatively, much more sophisticated user models but as far as we know, there are no such models nowadays). Assessing preferred reasons in the context of a specific application is a perspective for further work. We have an application in mind (a problem of sale prediction) where we could benefit from the help of an expert to evaluate the explanations that are generated.

Acknowledgments

Many thanks to the anonymous reviewers for their comments and insights. This work has benefited from the support of the AI Chair EXPEKCTATION (ANR-19-CHIA-0005-01) of the French National Research Agency. It was also partially supported by TAILOR, a project funded by EU Horizon 2020 research and innovation programme under GA No 952215.

References

[Adadi and Berrada, 2018] A. Adadi and M. Berrada. Peeking inside the black-box: A survey on explainable artificial intelligence (XAI). *IEEE Access*, 6:52138–52160, 2018.

[Ansótegui and Gabàs, 2017] C. Ansótegui and J. Gabàs. WPM3: an (in)complete algorithm for weighted partial MaxSAT. *Artificial Intelligence*, 250:37–57, 2017.

[Audemard *et al.*, 2022] G. Audemard, S. Bellart, L. Bounia, F. Koriche, J-M. Lagniez, and P. Marquis. Trading complexity for sparsity in random forest explanations. In *Proc. of AAAI'22*, 2022.

[Breiman *et al.*, 1984] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone. *Classification and Regression Trees*. Wadsworth, 1984.

[Breiman, 2001] L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[Brewka, 1989] G. Brewka. Preferred subtheories: an extended logical framework for default reasoning. In *Proc. of IJCAI'89*, pages 1043–1048, 1989.

[Darwiche and Hirth, 2020] A. Darwiche and A. Hirth. On the reasons behind decisions. In *Proc. of ECAI'20*, pages 712–720, 2020.

[Doshi-Velez and Kim, 2017] F. Doshi-Velez and B. Kim. A roadmap for a rigorous science of interpretability. *CoRR*, abs/1702.08608, 2017.

[Guidotti *et al.*, 2019] R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, and D. Pedreschi. A survey of methods for explaining black box models. *ACM Computing Surveys*, 51(5):93:1–93:42, 2019.

[Huang *et al.*, 2021] X. Huang, Y. Izza, A. Ignatiev, M. C. Cooper, N. Asher, and J. Marques-Silva. Efficient explanations for knowledge compilation languages. *CoRR*, abs/2107.01654, 2021.

[Ignatiev *et al.*, 2019] A. Ignatiev, N. Narodytska, and J. Marques-Silva. Abduction-based explanations for machine learning models. In *Proc. of AAAI'19*, pages 1511–1519, 2019.

[Ignatiev *et al.*, 2020] A. Ignatiev, N. Narodytska, N. Asher, and J. Marques-Silva. On relating 'why?' and 'why not?' explanations. *CoRR*, abs/2012.11067, 2020.

[Ignatiev, 2020] A. Ignatiev. Towards trustable explainable AI. In *Proc. of IJCAI'20*, pages 5154–5158, 2020.

[Izza and Marques-Silva, 2021] Y. Izza and J. Marques-Silva. On explaining random forests with SAT. In *Proc. of IJCAI'21*, pages 2584–2591, 2021.

[Lipton, 2018] Z. C. Lipton. The mythos of model interpretability. *Communications of the ACM*, 61(10):36–43, 2018.

[Lundberg and Lee, 2017] S. Lundberg and S-I. Lee. A unified approach to interpreting model predictions. In *Proc. of NIPS'17*, pages 4765–4774, 2017.

[Lundberg *et al.*, 2020] S. M. Lundberg, G. G. Erion, H. Chen, A. J. DeGrave, J. M. Prutkin, B. Nair, R. Katz, J. Himmelfarb, N. Bansal, and S. Lee. From local explanations to global understanding with explainable AI for trees. *Nat. Mach. Intell.*, 2(1):56–67, 2020.

[Martins *et al.*, 2014] R. Martins, V. M. Manquinho, and I. Lynce. Open-WBO: A modular MaxSAT solver. In *Proc. of SAT'14*, pages 438–445, 2014.

[Miller, 2019] T. Miller. Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence*, 267:1–38, 2019.

[Molnar, 2020] C. Molnar. *Interpretable Machine Learning*. Leanpub, 2020.

[Narayanan *et al.*, 2018] M. Narayanan, E. Chen, J. He, B. Kim, S. Gershman, and F. Doshi-Velez. How do humans understand explanations from machine learning systems? an evaluation of the human-interpretability of explanation. *CoRR*, abs/1802.00682, 2018.

[Pedregosa *et al.*, 2011] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[Quinlan, 1986] J. R. Quinlan. Induction of decision trees. *Machine Learning*, 1(1):81–106, 1986.

[Ribeiro *et al.*, 2016] M. T. Ribeiro, S. Singh, and C. Guestrin. "why should I trust you?": Explaining the predictions of any classifier. In *Proc. of SIGKDD'16*, pages 1135–1144, 2016.

[Ribeiro *et al.*, 2018] M. T. Ribeiro, S. Singh, and C. Guestrin. Anchors: High-precision model-agnostic explanations. In *Proc. of AAAI'18*, pages 1527–1535, 2018.

[Samek *et al.*, 2019] W. Samek, G. Montavon, A. Vedaldi, L.K. Hansen, and K.R. Müller. *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*. Springer, 2019.

[Shih *et al.*, 2018] A. Shih, A. Choi, and A. Darwiche. A symbolic approach to explaining bayesian network classifiers. In *Proc. of IJCAI'18*, pages 5103–5111, 2018.

[Srinivasan and Chander, 2020] R. Srinivasan and A. Chander. Explanation perspectives from the cognitive sciences - A survey. In *Proc. of IJCAI'20*, pages 4812–4818, 2020.

[Xu *et al.*, 2019] F. Xu, H. Uszkoreit, Y. Du, W. Fan, D. Zhao, and J. Zhu. Explainable AI: A brief survey on history, research areas, approaches and challenges. In *Proc. of NLPCC'19*, pages 563–574, 2019.